Performance of Likelihood-Based Estimation Methods for Multilevel Binary Regression Models

Marc Callens and Christophe Croux[1]

K.U. Leuven

Abstract: By means of a fractional factorial simulation experiment, we compare the performance of Penalised Quasi-Likelihood, Non-Adaptive Gaussian Quadrature and Adaptive Gaussian Quadrature in estimating parameters for multi-level logistic regression models. The comparison is done in terms of bias, mean squared error, numerical convergence, and computational efficiency. It turns out that, in terms of Mean Squared Error, standard versions of the Quadrature methods perform relatively poor in comparison with Penalized Quasi-Likelihood.

Keywords: Binary Regression, Fractional Factorial Experiment, Gaussian Quadrature, Monte Carlo Simulation, Multilevel Analysis, Penalised Quasi-Likelihood

[1] Dept. of Applied Economics, Katholieke Universiteit Leuven, Naamsestraat 69, B-3000 Leuven, Belgium, Email: christophe.croux@econ.kuleuven.ac.be

1. Introduction

Many social systems can be thought of as hierarchies (e.g. individuals in societies, students in research groups,…). The general idea is that individuals interact with the groups to which they belong: people are influenced by their contexts and vice versa. The main statistical problem of studying such hierarchical systems is the dependence of the data at the lower level of the hierarchical system, i.e. observations belonging to the same group will be dependent. Obviously, such dependencies violate the assumption of i.i.d. observations and hence, standard statistical analysis techniques are not appropriate.

In this paper we are interested in estimating the parameters of hierarchical regression models for correlated binary data. We thereby follow the random components approach (Raudenbush and Bryk, 2002; Snijders and Boskers, 1999). To fix ideas, suppose that we have a hierarchical system consisting of two levels: at the lower level we have individual data, belonging to different specified groups or clusters. The groups constitute the second level. The variable of interest is the binary variable y_{ij}, giving the response (zero or one) of the i th unit within the j th cluster. We denote by N the total number of clusters and by n_j the number of observations belonging to group j, i.e. the group size. The vector of explicative variables is given by x_{ij}, and gives information on the individual and/or the group to which it belongs. As is common in binary regression, one models the conditional probabilities $p_{ij} = \Pr(y_{ij} = 1 \mid x_{ij})$ via the model equation:

$$link(p_{ij}) = \alpha_j + \beta_j x_{ij}, \tag{1}$$

where $link$ is an increasing link function mapping the interval]0,1[on the real line. In the above model the intercepts α_j and regression coefficients β_j may contain random components. These random effects are typically assumed to follow a multivariate normal distribution with unknown means, variances and covariance.

Adequate multilevel analysis techniques have already been developed for linear regression models. However, for the analysis of multilevel non-linear models, much research is still ongoing. The basic problem is to obtain good estimates of the mar-

ginal distribution of the data, which takes the form of an intractable integral. Therefore, estimation has to proceed via approximation. We focus thereby on three different likelihood-based estimation procedures frequently used in the applied multilevel-modelling literature: Penalised Quasi-Likelihood (PQL), Non-Adaptive Gaussian Quadrature (NGQ) and Adaptive Gaussian Quadrature (AGQ). Their computing algorithms are standard implemented in the SAS macro GLIMMIX and procedure NLMIXED

According to experience of previous performance studies, who will be reviewed and referred to in Section 3, it is thought that the number of groups, the group sizes, the variance of the random effects, the average conditional probability and the size of the correlation between random effects all are influential factors for performance issues. The main objective of this study is to compare the performance of different estimation procedures for these varying conditions. The comparison is done in terms of the bias and the mean squared error of the estimators, but also computing time and convergence of the algorithm used to compute the estimator is taken into account.

In Section 2, we describe the model to be used in the performance study and discuss briefly the estimation methods. In Section 3, we review previous simulation studies. Focus in this paper is on a two-level binary regression model with random intercepts and slopes, possibly correlated, and fixed slopes for the level 2 and cross-level interaction term. Such a model is often appropriate in practice and several other simulation studies considered special cases of this model. Two link functions of practical interest are considered. The logit link has already been studied in several other performance studies, but rather in a fragmentary way and not always from a general perspective. The second link function we will consider is the complementary log-log link, which has not been the object of any performance study up to now. This cloglog link function is of great importance for multilevel discrete-time proportional hazard models. The latter models can be rewritten as multilevel binary regression models (Allison, 1995).

The design of our simulation study is described in Section 4. There is a lot of computational effort in computing the estimators. Hence, the design needs to be chosen with care to keep the total computational cost of the study under control. The simulation study was therefore run as a fractional factorial experiment. In this way, running simulations for a restricted number of sampling schemes can retrieve a maximum

amount of information. In Section 5, results for the different estimation procedures under varying sampling schemes are presented and discussed. Section 6 concludes the paper by summarizing the simulation results.

2. Multilevel Binary Regression

2.1. The Random Slope Binary Regression Model

Focus in this paper is on the two-level random slope binary regression model. The binary response depends on an observed covariate x_{ij} at the individual level, a covariate at the group level z_j and unobserved random effects $\boldsymbol{u_{0j}}$ and $\boldsymbol{u_{1j}}$ as follows:

$$link(p_{ij}) = \gamma_0 + \boldsymbol{u_{0j}} + (\gamma_1 + \boldsymbol{u_{1j}})x_{ij} + \gamma_2 z_j + \gamma_3 z_j x_{ij}, \quad (2)$$

where $j = 1,...,N$ and $i = 1,...,n_j$. In the simulation experiment, we only consider the case of equal group or cluster sizes: $n_j = n$ for every $j = 1,...,N$. To identify the parameters γ_0 and γ_1, we pose the restriction that the averages of the random effects equal zero. The random effects $\boldsymbol{u_j} = (\boldsymbol{u_{0j}}, \boldsymbol{u_{1j}})$ are independent and identically distributed for $j = 1,...,N$. The random intercept variance, $var(\boldsymbol{u_{0j}}) = \sigma^2_0$, the random slope variance, $var(\boldsymbol{u_{1j}}) = \sigma^2_1$, and the covariance between the two random effects $covar(\boldsymbol{u_{0j}}, \boldsymbol{u_{1j}}) = \sigma_{01}$, are called variance components. The magnitude of the variance components determines the degree of within-group correlation. The other 4 parameters of interest are the average intercept γ_0, the average slope γ_1, the cluster-level regression coefficient γ_2 and the cross-level interaction regression coefficient γ_3. If the latter coefficient equals zero, then the effect of the level 2 variable z_j on the conditional probabilities p_{ij} is independent of the value of the level 1 variables x_{ij}, and hence there is absence of interaction.

Two link functions are considered: the *logit* link and the *complementary loglog (cloglog)* link. A very popular specification in binary regression problems is the logistic regression model, where the *logit* link is taken:

$$link(p_{ij}) = \ln(p_{ij}/(1-p_{ij})). \quad (3)$$

The cloglog model is particularly important in the context of grouped-time survival analysis, where it is used to estimate the parameters of the Cox proportional hazard model (Allison, 1984). Here, the link function is given by the log of the negative log of the complement of the probability:

$$link(p_{ij}) = \ln(-\ln(1-p_{ij})). \quad (4)$$

Unlike the logit function, cloglog corresponds to an asymmetrical distribution. A *cloglog* transformed probability of 0.9 or 0.1 equals 0.83 and -2.25 respectively; a *logit* transformed probability of 0.9 or 0.1 gives -2.20 and +2.20 respectively.

Another well-known link function, mainly used in the field of econometrics, is the *Probit* link, given by the inverse of the cumulative distribution function of a standard normal. Since previous simulation studies focused mainly on the logit link, we did not included the Probit link in this study and limited the simulation study to two link functions: the logit corresponding to a symmetrical, and the cloglog corresponding to an asymmetrical distribution. We expect, however, that simulations results for the Probit link will be similar to those for the logit link.

2.2. Estimation Methods

Model (2) belongs to the class of Hierarchical Generalized Linear Models (HGLM). A general review of HGLM can be found in Raudenbush and Bryk (2002; chapter 10). We will now write up the likelihood for the random slope binary regression model (2). Let the response vector y consist of all the elements y_{ij}. These elements are, conditional on the random effects, supposed to be independent of each other, each element having a conditional density:

$$f_{Y_{ij}|\boldsymbol{u}_j}(y_{ij} \mid \boldsymbol{u}_j) \sim Bernoulli(p_{ij}). \quad (5)$$

The expected value of the Bernoulli distribution, which equals p_{ij}, is then, after applying the specified link function, modelled as a linear function of the covariates, see equation (2). One also needs to specify a distribution for the random effects.

The typical assumption is that $\boldsymbol{u_1},...,\boldsymbol{u}_N \sim N(0,\Sigma)$ are independent draws from a (multivariate) normal distribution of dimension *m*:

$$f(\boldsymbol{u}_j) \sim N(0,\Sigma), \tag{6}$$

where in our case $m = 2$ and

$$\Sigma = \begin{pmatrix} \sigma_0^2 & \sigma_{01} \\ \sigma_{01} & \sigma_1^2 \end{pmatrix}.$$

In order to estimate the parameters of such a HGLM one usually makes use of marginal maximum likelihood estimation. In this method, the marginal likelihood of the observed data, obtained by integrating out the distribution of the random effects, is maximised. From (5), and (6) we can write down a formula for the marginal likelihood L (conditional on the covariates):

$$L(y) = \int \prod_{j=1}^{N} \prod_{i=1}^{n_j} f_{Y_{ij}|\boldsymbol{u}_j}(y_{ij} \mid \boldsymbol{u}_j) f_{\boldsymbol{u}_j}(\boldsymbol{u}_j) d\boldsymbol{u}_j = \prod_{j=1}^{N} \int \prod_{i=1}^{n_j} f_{Y_{ij}|\boldsymbol{u}}(y_{ij} \mid \boldsymbol{u}) f_{\boldsymbol{u}}(\boldsymbol{u}) d\boldsymbol{u}, \tag{7}$$

where $L(y)$ depends on the unknown parameters $\gamma_0, \gamma_1, \gamma_2, \gamma_3, \sigma_0, \sigma_1, \sigma_{01}$. The likelihood (7), thanks to conditional independence, can thus be considered as a product of independent contributions from each of N clusters. Evaluating (7) requires the computation of N integrals of dimension m. The likelihood (7) needs then to be maximized with respect to the 7 unknown parameters of the model.

In general the integral (7) has no closed form and needs to be evaluated numerically. Maximisation of the likelihood proceeds then by standard methods such as the EM algorithm. Since the likelihood needs to be evaluated many times during the iterative maximization procedure, fast but reliable approximations to (7) are needed.

A number of effective ways to compute and maximise the likelihood have been developed: quasi-likelihood inference, numerical quadrature, Monte Carlo integration, stochastic approximation, simulated maximum likelihood,…In this paper, we compare the behaviour of one quasi-likelihood procedure: Penalised Quasi-Likelihood (PQL, Breslow and Clayton 1993) and two full likelihood approaches based on numerical integration: Non-Adaptive Gaussian Quadrature (NGQ, e.g. Aitkin 1999) and Adaptive Gaussian Quadrature (AGQ, Pinheiro and Bates 1995).

2.2.1. Penalised Quasi-Likelihood

In the Quasi-Likelihood approach, the density $f_{Y_{ij}|\boldsymbol{u}}(y_{ij} \mid \boldsymbol{u})$ is approximated by a multivariate normal. The resulting marginal distribution in (7) has then a closed form solution, and can be then directly maximised. When Taylor series expansions around the approximate posterior mode are used, this approach is called Penalised Quasi-Likelihood (Breslow and Clayton, 1993). A key feature of this method is that estimation proceeds by iteratively fitting linear mixed models.

A basic advantage of PQL over other computational methods for HGLMs is its computational efficiency. Therefore, PQL estimation is sometimes advocated as a starting value for other procedures and for exploratory reasons. Another advantage is that for complex models (e.g. having a large number of random effects and/or multiple hierarchies) the model may still be estimated by PQL, while other estimation methods fail.

However, the PQL approach deteriorates as the distribution of the response variable departs more from normality or if large variance components are present. The parameter estimates from PQL are then negatively biased (Breslow and Lin, 1995). Another disadvantage is that PQL does not directly involve the likelihood. So, this method cannot use likelihood-based inference such as likelihood ratio tests and likelihood based confidence intervals.

2.2.2. Gaussian Quadrature methods

An alternative approach is to approximate the integral (7) by numerical integration and then to maximize the likelihood with approximate values for the integrals. Numerical integration proceeds by Gauss-Hermite quadrature formula:

$$\int_{-\infty}^{\infty} h(v)e^{-v^2}dv \doteq \sum_{q=1}^{d} h(x_q)w_q \,, \tag{8}$$

where h is a smooth function. Here $x_1,...,x_d$ are the quadrature points, and $w_1,...,w_d$ the associated weights summing to one. The larger d, the number of quadrature points, the better the approximation in (8). For a given d, quadrature points and weights are tabulated. Note that, since the distribution of the random effects is

supposed to be normal, the N integrals appearing in (7) are of the above form. The estimator obtained by maximizing the likelihood approximated in this way is called the Nonadaptive Gaussian Quadrature (NGQ) estimator.

In the two-level multivariate random effects model (2), the N integrals are in fact double integrals, which are usually evaluated using Cartesian product quadrature (e.g., Bock and Aitkin, 1981). Then, the number of quadrature points in which the function h needs to be evaluated is of order d^2.

Gauss-Hermitian quadrature can be poor for functions that are not properly centered or non-smooth (McCulloch and Searle, 2001). A likely reason for this is that in these conditions, the cluster-specific integrands have very sharp peaks that may be located between adjacent quadrature points (Lesaffre and Spiessens, 2001). The performance of Gaussian Quadrature can be improved by integration methods that are called *adaptive* in the sense that they take into account the properties of the integrand. Such methods scale and translate the quadrature locations to place them under the peak of the integrand. In this way, the position of the quadrature points may vary from cluster to cluster. For more detail, we refer to Pinheiro and Bates (1995) who developed such an improvement over *nonadaptive* Gaussian Quadrature in the context of two-level random coefficient models. Since the quadrature points need to be scaled and translated, computing the approximations of the integral will be more time consuming for a fixed value of d. But, since the quadrature points will now be placed much more central in the region of interest, the approximation will be much more accurate, allowing for a smaller number of quadrature points. The resulting estimator will be called the Adaptive Gaussian Quadrature estimator (AGQ).

2.2.3. Alternative Methods

Besides PQL and quadrature methods, several other possibilities do exist. One alternative is to approximate the integral (8) with Monte Carlo averages. In this spirit, McCulloch (1997) compared Monte Carlo EM, Monte Carlo Newton-Raphson and Simulated Maximum Likelihood algorithms for approximating the Maximum Likelihood estimators. Most of these Monte-Carlo methods have not yet reached the stage of full development and are still the object of active research programmes (see Booth et al, 2001 for a review). The availability of these procedures is generally limited to

special-purpose research software that has not yet been fully tested for general use. Therefore, these methods are not included in the present study. We have chosen to make a comparison between different estimation procedures currently available in a major software package and well known to applied statisticians: PQL, NGQ and AGQ. Default implementation of these estimators will be used.

The difficulty in evaluating likelihoods for Hierarchical Generalized Linear Models has also led to the development of alternative estimation methods, not based on marginal maximum likelihood estimation, such as Bayesian Markov Chain Monte Carlo methods (Browne, 1998) and nonparametric maximum likelihood (Aitkin, 1999). The Bayesian approach has been implemented in the commercial software package MLwiN. Another approach consists in using the Generalized Estimation Equation methodogy (GEE, e.g. Diggle, Liang and Zeger 1994). Here one models the expected value and covariance of the marginal distribution directly. This approach leads to a loss in efficiency, since the GEE estimators are not the maximum likelihood estimators. Moreover, the parameters estimated by GEE are not directly comparable to those estimated by maximum likelihood.

3. Review of previous Simulation Studies

During the last decade or so, the performance of estimation methods for hierarchical generalized linear models has been the subject of several studies. Several papers proposing new estimation methods contain small-scale simulations studies (e.g., Goldstein and Rasbash, 1996). A smaller number of papers focus more specifically on performance comparison of different estimation procedures (e.g., Rodriguez and Goldman, 1995, 2001, Browne and Draper, 2000) in an experimental setting. But, often the performance of an estimator is only measured by its bias, and the number of simulation replications rarely exceeds 100. Furthermore, a number of papers are based on the so-called Rodriguez-Goldman data, named after their 1995 paper, and have been subject to criticism of being quite atypical due to the large random effects and small cluster sizes.

In this paper, a carefully selected experimental design is used to carry out a large-scale simulation study with 1000 replicates for every sampling scheme. The selected sampling schemes in the present study correspond to parameters values that we

believe to be quite representative. Moreover, exploiting the structure of the design it will be possible to quantify the effect of model parameters on performance indicators (see Section 4). We also look at several performance indicators: bias, mean squared error, convergence of the maximization routine, and computing time.

Most simulation studies limit their attention to simple models with only random intercepts. The performance of bivariate random effect models -including both a random intercept and a random slope- is far less documented. A possible explanation for this apparent lack of attention for more complex models might simply be related to the computational effort needed. Another prevailing preference in model choice is the almost exclusive focus on the *logit* link only. To our present knowledge, no simulation studies in the field of hierarchical generalized linear models have been concerned with asymmetrical link functions like the *cloglog*.

What conclusions can be drawn from studies that are concerned with multilevel binary regression models having both random slope and intercept? Only few such studies -such as Raudenbush et al. (2000) - are present in the literature. Moreover, these studies do not consider full random slope models (2), but reduced versions of (2) without the cluster-level term $\gamma_2 z_j$ and the cross-level interaction term $\gamma_3 z_j x_{ij}$. However, in multilevel models, these terms have an important interpretation, as they might be able to explain the variability of the slopes as well as the intercepts at the group level.

Raudenbush et al. (2000) consider a random slope binary regression model with parameter values close to those matching the Rodriguez-Goldman data, including asymmetric probabilities and correlated random effects. They conclude that PQL estimates are systematically underestimating true values. This negative bias is more prominent for the variance parameters compared to the regression parameters. The bias for Non-Adaptive Gaussian Quadrature and Adaptive Gaussian Quadrature was found to be much smaller. The precision of PQL, NGQ, and AGQ turned out not to differ much. The results in this paper, considering a much broader class of sampling schemes, will indeed confirm the biasedness of the PQL-procedure. But, on the whole, we found that the Mean Squared Error for the estimation of the variance components for PQL is substantially lower than for the quadrature methods.

4. Simulation Design

We are interested in estimating the seven parameters of the random slope binary regression model

$$link(p_{ij}) = \gamma_0 + \boldsymbol{u_{0j}} + (\gamma_1 + \boldsymbol{u_{1j}})x_{ij} + \gamma_2 z_j + \gamma_3 z_j x_{ij} .$$

The simulations are twofold, with the simulation study performed using a *logit* link function (3) and a *cloglog* link function (4). In both models, there are four regression parameters and three random components to estimate. The four regression parameters are the average intercept γ_0, the average slope γ_1, the cluster-level regression coefficient γ_2 and the cross-level interaction regression coefficient γ_3. The three random components are the random intercept variance σ^2_0, the random slope variance σ^2_1 and the intercept-slope covariance σ^2_{01}. Such a model can be thought of as a nested model where the dichotomous outcome is predicted, via a link function, by an individual level covariate, a cluster-level covariate and a cross-level interaction and where intercepts and slopes vary across clusters.

Seven factors, potentially affecting the performance measures, are varied in the simulation study: number of clusters (*A*), cluster size (*B*), size of the variance of the intercept random effect (*C*), size of the variance of the slope random effect (*D*), average conditional probability (*E*), size of the correlation between intercept and slope (*F*) and of course also the different estimation procedures (*M*: AGQ, NGQ and PQL). The first 6 factors *A*-*F* are related to sample size and parameter values and will be called the model factors. To study the effect of the model factors, each of them will be set at two different values: a low and a higher one. Table 1 presents an overview of the six model factors and their levels.

(Table 1, about here)

For a full factorial experiment, one needs to consider all possible combinations of the factor levels. With three methods and six binary design factors, we would need 3 x 2^6 = 192 different runs. A *run*, borrowed from the language of experimental design, corresponds here to a simulation experiment for one selected sampling distribution, as determined by a combination of the factor levels of *A*-*F*, and one selected estimator, as determined by the factor level of *M*. Carrying out two (one for each link function) such full factorial experiments at 1000 replications for every run would take an

estimated eight months to run on a modern computer. Hence, for computational efficiency we prefer to use a 3 x 2^{6-3}_{III} fractional factorial design. In a 2^{6-3}_{III} design, the number of runs is reduced from 2^6 to 2^3 at the price of being only of resolution three (see Wu, C.F and M. Hamada, 2000). A 3 x 2^{6-3}_{III} design allows us to estimate the main effects of the different factors on the performance of the estimator, while interaction terms between estimation method *M* and the model factors are not confounded with the main effects. Some interaction terms between model factors may, however, be confounded with main effects. In Table 2, the design matrix for the 2^{6-3}_{III} fractional part is presented.

(Table 2, about here)

In the data generation part of the experiment, for each of the 8 runs listed in Table 2, 1000 samples are generated from the model. For each simulated dataset, conditionally independent binary observations y_{ij} are generated within each cluster *j* with conditional response probabilities given by equation (2). The four possible combinations of the factors cluster size and number of clusters result in total sample sizes 150, 500, 3000 and 10000 respectively. Covariate values x_{ij} and z_j are generated from independent standard normal distributions. The values of the regression parameters γ_1, γ_2 and γ_3 are set equal to 1. The values for the average intercept γ_0 are set to -2.1972 or 0 for a *logit* model and -2.2504 or -0.3665 for a *cloglog* model, corresponding to small (0.1) or central (0.5) average conditional probabilities. The random effects $\boldsymbol{u_{0j}}$ and $\boldsymbol{u_{1j}}$ are generated by a bivariate normal distribution, with zero mean and covariance matrix according to the values of the factors *C, D*, and *F*. All random numbers are generated by SAS RANUNI and RANNOR functions, which are based on a congruential generator (Fan et al., 2002). The 7 model parameters of interest are then estimated by the three different approaches for each of the 2 x 8000 such simulated datasets. Recall that the analysis is done once for the *logit* model and once for the *cloglog* model. This keeps a Wintel PC, with a Pentium IV CPU running at 2.000 MHZ, busy for about a month.

Both Gaussian quadrature methods (NGQ and AGQ) are run on SAS Version 8.1 using PROC NLMIXED (Wolfinger, 1999). These algorithms select the number of quadrature points (*q*-points) such that the likelihood value yields a negligible differ-

ence if the next higher number of *q*-points would be used. As starting values for the parameters, true parameters values are used. The default maximisation routine in NLMIXED is Dual Quasi-Newton. To apply the method of Penalised Quasi-Likelihood (PQL), we use the GLIMMIX SAS-macro (Littell et al., 1996). The estimating algorithm iteratively fits a linear mixed model (by repeatedly calling SAS PROC MIXED) to a pseudo response. By default, PROC MIXED uses restricted maximum likelihood estimation (REML).

The three estimation methods are evaluated at four performance dimensions: numerical convergence, bias, mean squared error and computation time. Numerical convergence is measured by the convergence rate. This convergence rate is based on the indicator variables produced by the macro GLIMMIX and PROC NLMIXED to confirm whether numerical convergence has been reached or not. Output from the SAS routines was gathered using the output delivery system (ods), standard available in SAS version 8.1.

For a given univariate population parameter θ (of which the value is known in the context of a simulation study) and corresponding estimates $\hat{\theta}_1,...,\hat{\theta}_M$ from *M* independent replications, the Monte Carlo estimate of bias is computed as

$$BIAS = \frac{1}{M}\sum_{i=1}^{M}\hat{\theta}_i - \theta \tag{9}$$

and the Monte Carlo estimate of the Mean Squared Error (MSE) is:

$$MSE = \frac{1}{M}\sum_{i=1}^{M}(\hat{\theta}_i - \theta)^2 . \tag{10}$$

Finally, computational efficiency is measured as CPU time for computing one single estimate. As GLIMMIX is actually written as a SAS macro and the two other programs are compiled versions, the comparison between PQL and NGQ and AGQ is not completely fair, with PQL being disadvantaged.

For each of the simulation schemes (each one corresponding to different settings of the model factors, as described in Table 1) the performance indicators were computed for the 3 methods. This is illustrated in Table 3, where the convergence probabilities are given for each of the 8 different simulation runs and for every method. We already clearly see that PQL gives the best results for this indicator. To explore the effects of the different factors, we will now use the fractional factorial design to our advantage.

Denote by $y_{mabcdef}$ the performance indicator for method m (where m = AGQ, NGQ, or PQL) computed for a simulation run where the factors A, B, C, D, E, F were set at the levels a, b, c, d, e, f. The factor levels a, b, c, d, e, f are coded as $+$ or $-$.

The factorial ANOVA model we use is then given by:

$$y_{mabcdef} = \mu_m + \alpha_a^A + \alpha_b^B + \alpha_c^C + \alpha_d^D + \alpha_e^E + \alpha_f^F + \varepsilon_{mabcdef} \quad (11)$$

In the above equation, we have the restriction that α_+^A = -α_-^A, α_+^B = -α_-^B, etc. Primary interest is in the quantities μ_{AGQ}, μ_{NGQ}, and μ_{PQL}, which are an overall performance measure of the method over the different simulation schemes. Furthermore, the parameter α_+^A gives the effect of changing factor A from level $-$. to level $+$, and it will simply be called the effect of factor A. Similar of course for the effects of other binary factors B, C, D, E, F.

In Table 4, using the convergence probability as performance indicator, we first report the *grand mean*, being nothing but the average of the performance indicators for the 3 methods over all simulation runs. Then the estimates of the quantities, μ_{AGQ}, μ_{NGQ}, and μ_{PQL} are reported, followed by the estimates of the factor effects α_+^A, α_+^B, etc. Subsequent tables present the bias, MSE, and computation time results from the ANOVA estimation.

5. Results

5.1. Numerical convergence

Consider as performance indicator the percentage of times that the iterative computation of the estimator converged. ANOVA estimation of the model (11) yields the results in Table 4. As a general comment, one immediately sees that convergence problems occur very frequently. The algorithms are less reliable for *cloglog* models compared to *logit*. From an applied viewpoint, this implies that the user frequently will need to change starting values, number of iterations steps, or other tuning parameters manually, in order to get convergence of the numerical procedures. Allow-

ing for changes of the tuning parameters of the numerical routines, can alter the numbers in Table 4 substantially, hence we do not want to make any overstatements here regarding the relative performance of the methods.

(Table 4 about here)

Factor *B*, Cluster size, has an important positive effect on convergence. The larger the cluster size, the higher the probability that the algorithm will converge. All other factors are much less important. For example, the magnitude of the intercept and slope variances, factors *C* and *D,* will not affect much the convergence properties of the algorithm.

Note that, when computing the other performance indicators Bias, MSE, and computing time, only samples for which all three estimation procedures converged were used.

5.2. Bias

5.2.1. Regression Parameters

In Table 5, results for the biases of the estimates for the parameters γ_0, γ_1, γ_2 and γ_3 are given. Standard errors around the reported numbers are about 0.007. Recalling that the true parameter values are all 1, we see that the biases for the quadrature methods are negligible, while those for PQL are more substantive. PQL overestimates the average of the random intercept γ_0, but systematically underestimates the three regression coefficients γ_1, γ_2 and γ_3. Note that the bias for the fixed effects parameters γ_2 and γ_3 is of the same order of magnitude as for the expected value γ_1 of the random coefficients. These findings hold for both link function, although the bias for PQL is slightly higher for the asymmetric *cloglog* function.

(Table 5 about here)

Regarding the effect of the model factors, the most important seems to be the variance of the random slope (*D*) and the correlation between random slope and intercept (*F*).

5.2.2. Variance Components

Results for the estimates of the random components are in Table 6. Standard errors around the reported numbers are now around 0.015. Biases are now clearly present for all three methods, but are less pronounced for AGQ and NGQ. Note that bias for the covariance parameter is negligible for AGQ and NGQ, and of a small order for PQL. The bias for the random components decreases further (in absolute values) for the *cloglog* model using quadrature methods, but the opposite happens for PQL.

(Table 6 about here)

For the bias of the estimator of the variance component of the random intercept term the most influential design factor is the intercept variance (*C*). For the bias of the estimator of the variance component of the random slope term the most influential design factors are the slope variance (*D*) and the intercept-slope correlation (*F*). Cluster size (*B*) affects the average estimates of the slope variance component more than the intercept variance component.

To conclude the study of the bias of the different procedures, we can say that using quadrature procedures yields negligible biases for the estimates of the regression parameters, very small biases for the intercept variance estimate, but larger biases for the slope variance components (at least for the *logit* link). It is also confirmed that the bias for PQL is more important.

5.3. Mean Squared Error

5.3.1. Regression Parameters

In Table 7, results for the ANOVA analysis of the Mean Squared Error for the regression parameters are presented. The overall picture is that the MSEs of the three estimators are comparable over the different sampling schemes (standard errors around the reported numbers are again of the order 0.007). This observation holds for both link functions, giving strikingly similar results.

(Table 7 about here)

The design factors *A* and *B*, being related to the sample size, have of course a negative effect on the MSE. Increasing the variance of the random effects makes precise estimation more difficult, hence *C* and *D* let MSE increase. Furthermore, presence of correlation between the random effects (factor *F*) makes the MSE decrease. Presence of correlation means that knowledge on one random effect gives information on the other, hence makes estimation easier. Finally, factor *E* has consistently a negative impact. If samples are balanced, i.e. having about as many one as zero binary outcomes, estimation will be more precise.

5.3.2. Variance Components

In Table 8, the analysis of the MSE of the estimates of the random effects σ^2_0, σ^2_1 and σ_{01} is presented. These results are most interesting. The MSE of the PQL estimates are much smaller than the corresponding MSE of the quadrature methods, both having similar MSEs. For the *logit* link the MSE decreases almost by a factor of two when using PQL, for the *cloglog* link the difference in MSE is somewhat less but still significant.

(Table 8 about here)

The effect of the model factors *A*-*F* proceeds in exactly the same direction as for Table 7, and the same discussion applies.

The simulation study shows that although PQL gives rise to larger biases for the variance components, its MSE is still outperforming that of the quadrature methods. Note that the use of biased, but more precise, estimators is not uncommon in statistics, e.g. ridge regression.

The fact that both Quadrature methods perform unexpectedly poor compared to PQL might be explained by the fact that the number of selected quadrature points d was not adequate. Theoretically, when the number of quadrature points tends to infinity, AGQ and NGQ work with the exact likelihood and should then be equal to the true Maximum Likelihood estimator. Since the data are generated according to the model,

and since the sample sizes are quite large, the latter estimator should be the most precise. It is quite unclear how large exactly one has to choose d before the MSE of the quadrature methods becomes better, but one should not forget that taking d too large leads to impossibly large computing times.

5.4. Computation time

As a last performance indicator, computing time of the different estimation procedures, as implemented in SAS, is studied (Table 9). NGQ is by far the slowest method. Depending on the link function begin used, AGQ or PQL is fastest, but the difference is not very large. However, one has to take into account that PQL is run as a SAS macro, which slows down execution time considerably. Hence we can safely stat that PQL is by far preferable in computing time.

Note that both quadrature methods have more difficulty in estimating *cloglog* models than *logit* models. It is instructive to look at the average number of quadrature points needed. For *logit* models the average value of d equals 14 for AGQ and 126 for NGQ. For *cloglog* models the average value of d amounts to 24 for AGQ and to 168 for NGQ. This illustrates that much more quadrature points need to be taken before NGQ yields a good approximation of the integrals, resulting in a much higher total computation time than for AGQ method.

(Table 9 about here)

From Table 9 one sees that changing the level of a model factor from - to + , results in increased computation time. For factors *A , B, C* and *D* this is logical. While having correlation between the random effects made the MSE decrease, it increases the computation time. The same remark applies for factor *E*, somehow surprisingly.

6. Conclusion

This paper compared the performance of three estimation methods for multilevel binary regression models: Penalised Quasi-Likelihood (PQL), Non-Adaptive Gaussian Quadrature (NGQ) and Adaptive Gaussian Quadrature (AGQ). These likelihood-

based methods are frequently used in the applied multilevel-modelling literature to estimate multilevel logistic regression and discrete-time proportional hazard models. Standard implementation of SAS GLIMMIX and PROC NLMIXED was used to actually perform the calculations.

Different sampling schemes were selected according to a fractional factorial design. Hereby we could reduce the total computing cost of the simulation study, while still retrieving enough information on performance of the estimators under a variety of different circumstances. Moreover, the fractional factorial design of the simulation experiment allows quantifying the effect of different model parameters on the performance of the estimators. Bias, Mean Squared Error, computing time, and convergence of the estimation routine were considered as performance measures.

Comparing the quadrature methods yields close results with respect to Bias and Mean Squared Error, but the Non-Adaptive version was by far the slowest. Hence, it is confirmed that AGQ is to be preferred above NGQ. Comparing PQL with AGQ showed that the bias was larger for PQL, hereby confirming previous studies which mainly focused on the bias. However, PQL gave the most precise estimates, as measured by the MSE. These conclusions hold both for multilevel logistic regression (*logit* link) and the proportional hazard model (*cloglog* link) we considered.

Three main messages can be stated after having carried out this large-scale simulation study: (i) convergence problems arise very frequently when executing standard programs to estimate multi-level binary regression models, even when the starting values equal the true parameter values. Development of safer, fully automatic, estimation procedures is required; (ii) PQL is a fast estimation method, and although biased, it beats Quadrature methods in terms of Mean Squared Errors. (iii) Automatic selection of the number of quadrature points in AGQ might be inadequate and lead to a loss in MSE.

Most important conclusion of this simulation experiment is that although Penalized Quasi-Likelihood suffers from a larger bias (confirming previous findings in the literature), it performs better in terms of Mean Squared Error than standard versions of the Quadrature methods.

Acknowledgements

This research was supported by a FWO grant *Bijzondere doctoraatsbeurs 2002-2003.* We are grateful to Peter Goos (KULeuven) for his advice on experimental design issues. We also want to thank Oliver Schabenberger (SAS Institute Inc), developer of NLMIXED and GLIMMIX for his valuable hints.

References

Aitkin, F. (1999), A general maximum likelihood analysis of variance components in generalized linear models, *Biometrics*, 55, pp. 117-128.

Allison, P.D. (1984), *Event history analysis,* Beverly Hills: Sage.

Allison, P.D. (1995), *Survival Analysis using the SAS System: a practical guide*, Cary: SAS Institute Inc.

Bock, R.D. and M. Aitkin (1981), Marginal maximum likelihood estimation of item parameters: application of the EM algorithm, *Psychometrika*, 46, pp. 443-459.

Booth, J.G., Hobert, J.P. and W. Jank (2001), A survey of Monte Carlo Algorithms for maximizing the likelihood of a two-stage hierarchical model*, Statistical Modeling*, 1, pp. 333-349.

Breslow, N. and D. Clayton (1993), Approximate inference in generalized linear models, *Journal of the American Statistical Association*, 88, pp. 9-25.

Breslow, N. and X. Lin (1995), Bias correction in generalized linear mixed models, *Biometrika*, 88, pp. 81-91.

Browne, W. J. (1998), *Applying MCMC Methods to Multilevel Models*, PHD dissertation, Department of Mathematical Sciences, University of Bath.

Browne, W.J. and D. Draper (2000), Implementation and performance issues in the Bayesian fitting of multilevel models, *Computational Statistics*, 15, pp. 319-420.

Diggle, P. J., Liang, K.-Y., and Zeger, S. L. (1994), *Analysis of Longitudinal Data*, Oxford, U.K.: Oxford University Press

Fan, X., Felsovalyi, A., Sivo S.A. and S.C. Keenan (2002), *SAS for Monte Carlo Studies, A Guide for Quantitative Researchers*, Cary: Sas Institute.

Goldstein, H. and J. Rasbash (1996), Improved approximations for multilevel models with binary responses, *Journal of the Royal Statistical Society A*, 159, pp. 505-513.

Lesaffre, E. and B. Spiessens (2001), On the effect of the number of quadrature points in a logistic random-effects model: an example, *Applied Statistics*, 50, pp. 325-335.

Littell, R.C., Milliken G.A., Stroup W.W. and R.D. Wolfinger (1996), *SAS System for Mixed Models*, Cary: Sas Institute.

McCulloch, C. E. (1997), Maximum Likelihood Algorithms for Generalized Linear Mixed Modes, *Journal of the American Statistical Association*, 92, pp.162-170.

McCulloch, C.E. and S.R. Searle (2001), *Generalized, Linear and Mixed Models*, New York: Wiley.

Pinheiro, J. and D. Bates (1995), Approximations to the log-likelihood function in the nonlinear mixed-effects model, *Journal of Computational and Graphical Statistic*s, 4(1), pp. 12-35.

Raudenbush, S.W., Yang, M. and M. Yosef (2000), Maximum Likelihood for hierarchical models via high-order, multivariate Laplace approximation, *Journal of Computational and Graphical Statistics*, 9, pp. 141-157.

Raudenbush S.W. and A.S. Bryk (2002), *Hierarchical Linear Models*, Thousand Oaks: Sage.

Rodriguez, G. and N. Goldman (1995), An assessment of estimation procedures for multilevel models with binary responses, *Journal of the Royal Statistical Society A*, 158, pp. 73-89.

Rodriguez, G. and N. Goldman (2001), Improved estimation procedures for multilevel models with binary response: a case-study, *Journal of the Royal Statistical Society A*, 164, pp. 339-355.

Snijders, T.A.B. and R.J. Boskers (1999), *Multilevel Analysis, An introduction to Basic and Advanced Multilevel Modeling*, London: Sage Publications.

Wolfinger, R.D. (1999), *Fitting Nonlinear Mixed Models with the new NLMIXED Procedure*, Proceedings of the 24th Annual SAS Users Group International Conference (SUGI 24), pp. 278-284.

Wu, C.F. and M. Hamada (2000*), Experiments, Planning, Analysis and Parameter Design Optimization*, New York: Wiley.

Table1 : Levels of the model factors in the simulation experiment

Factor	*Level*	
	-	+
A: number of clusters	30	100
B: cluster size	5	100
C: intercept variance	0.1	1
D: slope variance	0.1	1
E: average probability	0.1	0.5
F: intercept-slope correlation	0	0.5

Table 2: Design matrix for the 2^{6-3}_{III} *fractional factorial part of the experiment*

Run	*Factor*					
	A	*B*	*C*	*D*	*E*	*F*
1	-	-	-	-	+	+
2	-	-	+	+	-	-
3	-	+	-	+	-	+
4	-	+	+	-	+	-
5	+	-	-	+	+	-
6	+	-	+	-	-	+
7	+	+	-	-	-	-
8	+	+	+	+	+	+

Table 3: Convergence probability (%) for every experimental run by estimation method (logit link)

	Method		
Run	*AGQ*	*NGQ*	*PQL*
1	53.2	58.9	83.9
2	42.6	66.3	75.7
3	91.2	93.4	99.4
4	98.8	98.1	99.9
5	75.5	84.4	100.0
6	49.8	68.4	79.2
7	99.3	99.6	100.0
8	100.0	99.5	100.0

Table 4: Convergence probability (%) when estimating model (2). Results are given for AGQ, NGQ and PQL, together with effects of the model factors, both for the logit and cloglog link

	Logit	Cloglog
Grand Mean	84.0	71.5
Mean AGQ	76.3	74.8
Mean NGQ	83.5	77.3
Mean PQL	92.2	62.4
Effect A (number of clusters)	3.9	3.5
Effect B (cluster size)	14.2	19.1
Effect C (intercept variance)	-2.5	-1.5
Effect D (slope variance)	1.6	-1.0
Effect E (average probability)	3.6	-3.1
Effect F (intercept-slope corr)	-2.6	-1.3

Table 5: BIAS of the estimator of the average random intercept γ_0, the average random slope γ_1, and the fixed regression parameters γ_2 and γ_3 of model (2). Results are given for AGQ, NGQ and PQL together with effects of the model factors, both for logit and cloglog link

	Logit				Cloglog			
	γ_0	γ_1	γ_2	γ_3	γ_0	γ_1	γ_2	γ_3
Grand mean	0.010	0.004	0.008	0.001	0.033	-0.024	-0.014	-0.022
Mean AGQ	-0.021	0.033	0.034	0.032	-0.009	0.002	0.015	0.011
Mean NGQ	-0.018	0.032	0.031	0.027	-0.005	0.001	0.008	0.011
Mean PQL	0.073	-0.051	-0.040	-0.054	0.114	-0.077	-0.067	-0.089
Effect A	-0.007	-0.007	-0.012	-0.019	-0.006	-0.004	-0.009	-0.008
Effect B	-0.006	-0.011	-0.020	-0.008	-0.021	0.017	0.005	0.016
Effect C	0.003	-0.009	-0.009	-0.020	0.020	-0.004	-0.019	-0.017
Effect D	0.007	-0.028	-0.006	-0.018	0.012	-0.031	-0.020	-0.012
Effect E	-0.011	0.002	0.003	0.018	-0.021	0.000	0.011	0.010
Effect F	-0.006	0.029	0.007	0.013	-0.011	0.030	0.015	0.009

Table 6: BIAS of the variances of the random intercepts, random slopes, and the covariance between them, as in model (2). Results are given for AGQ, NGQ and PQL together with effects of the model factors, both for logit and cloglog link.

	Logit			Cloglog		
	σ^2_0	σ^2_1	σ_{01}	σ^2_0	σ^2_1	σ_{01}
Grand mean	0.000	0.011	-0.022	-0.023	-0.051	-0.025
Mean AGQ	0.041	0.059	-0.018	0.013	-0.007	-0.012
Mean NGQ	0.044	0.071	-0.019	0.022	0.011	-0.014
Mean PQL	-0.083	-0.097	-0.029	-0.106	-0.158	-0.049
Effect A	-0.001	0.000	-0.011	-0.010	0.009	-0.003
Effect B	-0.022	-0.035	0.012	-0.001	0.031	0.016
Effect C	-0.045	0.008	-0.019	-0.021	-0.010	-0.017
Effect D	-0.015	-0.066	0.007	-0.000	-0.088	-0.000
Effect E	0.025	-0.006	0.016	-0.001	0.006	0.011
Effect F	0.017	0.047	-0.016	-0.002	0.071	-0.011

Table 7: MSE of the estimator of the average random intercept γ_0, the average random slope γ_1, and the fixed regression parameters γ_2 and γ_3 of model (2). Results are given for AGQ, NGQ and PQL together with effects of the model factors, both for logit and cloglog link.

	Logit				Cloglog			
	γ_0	γ_1	γ_2	γ_3	γ_0	γ_1	γ_2	γ_3
Grand mean	0.055	0.065	0.056	0.077	0.054	0.057	0.044	0.064
Mean AGQ	0.057	0.070	0.061	0.084	0.050	0.057	0.045	0.063
Mean NGQ	0.054	0.069	0.059	0.079	0.049	0.057	0.046	0.061
Mean PQL	0.055	0.056	0.048	0.069	0.064	0.056	0.043	0.067
Effect A	-0.025	-0.036	-0.033	-0.045	-0.023	-0.029	-0.023	-0.032
Effect B	-0.040	-0.048	-0.038	-0.058	-0.040	-0.042	-0.028	-0.045
Effect C	0.035	0.015	0.022	0.014	0.042	0.003	0.020	0.012
Effect D	0.009	0.022	0.009	0.024	0.012	0.017	0.006	0.016
Effect E	-0.027	-0.022	-0.011	-0.021	-0.034	-0.008	-0.011	-0.019
Effect F	-0.013	-0.011	-0.014	-0.012	-0.016	-0.007	-0.010	-0.004

Table 8: MSE of the variances of the random intercepts, random slopes, and the covariance between them, as in model (2). Results are given for AGQ, NGQ and PQL together with effects of the model factors, both for logit and cloglog link.

	Logit			Cloglog		
	σ^2_0	σ_{u1}	σ_{01}	σ^2_0	σ^2_1	σ_{01}
Grand mean	0.138	0.177	0.061	0.108	0.123	0.041
Mean AGQ	0.163	0.200	0.077	0.127	0.120	0.050
Mean NGQ	0.156	0.213	0.071	0.125	0.130	0.051
Mean PQL	0.094	0.117	0.036	0.072	0.120	0.023
Effect A	-0.043	-0.059	-0.028	-0.042	-0.050	-0.021
Effect B	-0.109	-0.144	-0.052	-0.081	-0.091	-0.032
Effect C	0.103	0.037	0.040	0.095	0.037	0.034
Effect D	0.013	0.103	0.028	0.024	0.095	0.024
Effect E	-0.077	-0.053	-0.036	-0.070	-0.053	-0.030
Effect F	-0.027	-0.072	-0.024	-0.036	-0.065	-0.019

Table 9: Computation Times (in seconds) for estimating model (2). Results are given for AGQ, NGQ, and PQL, together with effects of the model factors, both for logit and cloglog link.

	Logit	Cloglog
Grand mean	32.3	46.9
Mean AGQ	12.0	17.3
Mean NGQ	67.9	109.8
Mean PQL	16.9	13.6
Effect A	14.6	20.9
Effect B	28.2	40.4
Effect C	14.3	25.7
Effect D	10.9	22.4
Effect E	14.0	25.9
Effect F	10.8	21.9